

Voronoi 算法模型及其程序实现

傅廷亮¹,尹雪涛¹,张扬²

(1. 中国科学技术大学计算机系,安徽 合肥,230027;

2. 北京西陆信息技术有限公司,北京 100086)

摘要:在自然科学和社会科学的诸多研究领域的问题中,Voronoi(沃洛诺依)多边形网络常常被用来做为计算机仿真模型。由于按照定义形成 Voronoi 多边形网络较难用程序实现,人们多采用 Delauney(狄洛尼)三角形的方法完成模型的构建。该文首先介绍 Voronoi 多边形和 Delauney 三角形的定义和形成模型的方法,然后就 Voronoi 多边形的计算机程序实现进行讨论,模型中使用少量的多边形和周期边界条件可以忽略系统边界效应的影响,最后给出了用 Turbo C 和 C++ 所写程序的基本框架。

关键词:计算机仿真;沃洛诺依多边形网络;狄洛尼三角形

中图分类号:TP202+.4 **文献标识码:**B

Voronoi Algorithm Model and the Realization of Its Program

FU Ting-liang¹, YIN Xue-tao¹, ZHANG Yang²

(1. Dept. of Computer, USTC, Hefei Anhui 230027, China;

2. Xilu Information Technology Co., Ltd, Beijing 100086, China)

ABSTRACT: The Voronoi polygon network is used usually as a computer simulation model in research fields of natural and social sciences. It is difficult to achieve program design for the Voronoi definition. Researchers like to use Delauney triangle to build their computer program model. This paper introduces the principle of the Voronoi polygon and the Delauney triangle and their forming method first, then discusses their program realization. These initial network models have 100 - 400 polygons and periodic boundary conditions so the boundary effects can be neglected, and finally the frame of Turbo C and C++ program is provided.

KEYWORDS: Computer simulation; Voronoi polygon network; Delauney triangle

1 引言

在自然科学和社会科学的几乎所有研究领域中,都能找到 Voronoi 多边形网络的应用的例子(见文献[2], [3], [5])。Voronoi 多边形网络的基本思想是为点创建区域,一个 Voronoi 多边形可认为是一个点所影响的区域。例如在生态学中,常常需要研究某种生物体的幸存者依赖于邻居的个数,它一定要为食物和光线而竞争;森林中树木种类和动物的 Voronoi 图被用来研究太拥挤产生的后果。微观世界中的分子结构是受各种力的综合影响的结果;为销售人员创建管辖区;寻找零售总经销区域;基于加权 Voronoi 图的城市影响范围划分;计算流体力学中常用的网格生成;二维细胞结构的计算机仿真模型;地理信息系统中的地图数据库和计算机

辅助设计系统等诸多领域都能看到 Voronoi 多边形网络的身影。

目前研究人员使用计算机仿真技术研究自然界各领域中的难解问题已成为必然的趋势,通过建立仿真模型,使其按某类事物本身的内在运动规律产生演变,就可以在计算机上看到该类事物的运动图象,身临其境的感受到其中的奥秘和乐趣。计算机仿真模型的构建质量如何直接影响仿真的结果,这里所用到的模型又分为多种类型,作者曾在国外参加过 Voronoi 多边形网络的应用研究课题,对构成模型的多种方法均有涉猎(见文献[1]),那时大多使用 Fortran 77 做为建模和仿真语言,而仿真结果也大多通过打印机或绘图仪输出,调试程序和观察系统动态行为很不方便,相比之下,目前广泛使用的计算机可视化语言要好得多,例如我们使用了 C++ Builder4.0 语言构建二维 Voronoi 多边形模型。本文只讨论几何法形成的二维 Voronoi 多边形网络模型,它已被广泛应用于各领域计算机仿真中做为初始模型(见文献[4])。

2 Voronoi 网络图和 Delauney 三角形

图1所示的Voronoi图又叫泰森多边形或Dirichlet图,它由一组连续多边形组成,多边形的边界是由连接两邻点线段的垂直平分线组成。

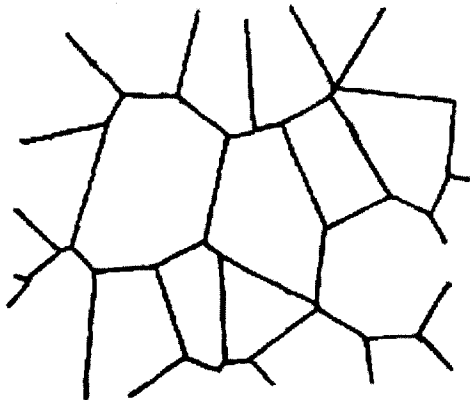


图1 一个Voronoi多边形网络(选自文献[1])

目前产生二维Voronoi网络的方法有以下两种,一种称为几何法,也就是精确模型法。另一种为细胞生长法,细胞生长法多用于蒙特卡罗法计算机仿真中。本文只讨论使用几何法构建模型,首先介绍根据二维Voronoi网络的定义完成模型,然后讨论使用Delauney(狄洛尼)三角形构成网络的方法。

根据二维Voronoi网络的定义建构模型的过程如下:

1) 首先在平面上布下 n 个有区别的 P_1, P_2, \dots, P_n ,这些点为面状分布,不都在同一条直线上,我们把这 n 个点称为原始点,这些原始点由随机数产生,这 n 个点便成为 n 个Voronoi多边形的中心。

2) 选定某一原始点作为参考点,以该点为起点做与其它 $n-1$ 个原始点的连线,再作这 $n-1$ 根连线的垂直平分线,这些垂直平分线必然相交,构成一些多边形,但是只有那些离参考点最近的垂直平分线围成的多边形才是所需要的,这个多边形内只有一个原始参考点,它不包含任何其它的原始点。重复以上过程,依次用其它 $n-1$ 个原始点做参考点形成 $n-1$ 个多边形,从而构成一个二维随机结构Voronoi网络模型。

从以上描述可见,由于程序中要计算点间连线和垂直平分线的交点,并要保证这些垂直平分线围成的多边形内只包含一个原始参考点,且要保存和判断的因素过多,所以这种方法实现起来较为困难,而使用Delauney三角形建模则要容易一些。用Delauney三角形构成网络的方法如下:

1) 同以上Voronoi网络定义建构模型的过程1)。

2) 先选定某一个原始点(图2中a点)作为参考点,再找与其邻近的两个原始点(b点和c点)组成一个Delauney三角形,再找一个原始点d与ac边组成另一个Delauney三角形,

这两个三角形(称为双三角形)以ac边为公共边。依次以a点为参考点作其它Delauney三角形,图1中有5个三角形以a点为中心。依次以b点、c点以及其它 $n-1$ 个参考点为中心作出全部Delauney三角形。每个Delauney三角形内部不包含任何原始点。

3) 分别作每个Delauney三角形三条边的垂直平分线,计算出它们的中心,相邻的双三角形的两个中心点的连线就是以a点为核心的Voronoi多边形的一条边。所有三角形的中心点连线组成一个有 n 个多边形的Voronoi网络。相比之下,这种方法既简明又易于程序实现,计算量也小得多。

Delauney三角形有如下特点:

Delauney三角形的三个顶点是相邻的三个原始点,任何一个Delauney三角形的外接圆的内部不能包含任何原始点,也不包含除此三角形本身的三个顶点以外的任何结点,并且此三角形元素的外接圆与内切圆的半径之比满足一定的关系,即如果一个三角形满足此条件那就是Delauney三角形。Delauney三角形的另一个特点是最大化最小角原则:每两个相邻的三角形构成的凸四边形的对角线,在相互交换后,六个内角的最小角不再增大。由这些三角形形成的三角形网格总是具有最优化特征。虽然在形成原始点时按一定顺序先后产生,但在形成Delauney三角形时可从任何原始点开始,所形成的Delauney三角网络都是相同的,即Delauney三角形形成的网络是唯一的,由此产生的Voronoi多边形网络也是唯一的。

Delauney三角形和由此产生的Voronoi网络如图2所示。

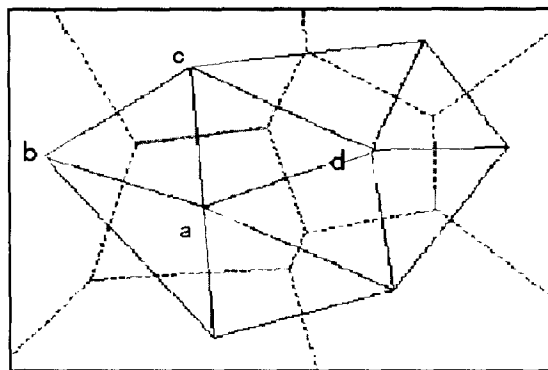


图2 Delauney三角网与Voronoi图

3 Delauney 三角形和 Voronoi 图的程序实现

在以前的计算机仿真中,大多采用基于过程的程序设计方法。这是因为大多数仿真问题需要用精确的时序关系来保证程序的正常运行。本模型也需要时序关系的保证,比如说,Voronoi网络的创建就必须在有效实施一些计算后完成,Delauney三角形的三个顶点和Voronoi多边形数据的采集必须在某些操作(如:计算)之后进行。

这种纯粹的面向过程的设计对于某些只对时序要求较强,而对对象属性的分析较弱的场合还是比较适合的,但是,

只单纯地考虑时序关系,而弱化所处理对象本身特征的过程已不多见,更多的时候是要求我们在某一指定时刻或某一指定地点对对象属性进行研究,大部分过程也是为了改善或增加,删除属性,以期得到预想中的性质。

本文中仿真模型最早用于二维肥皂泡的计算机仿真中,该模型的研究按编程语言来分已经经历了三代:Fortran 语言、Turbo C 和 C++ 等。这里介绍的总体设计思想是面向对象的,这个对象可以说就是整个仿真工程,它包括 Delauney 三角形的三个顶点和 Voronoi 多边形顶点的每个坐标点属性,还有其它一些多边形的属性(例如面积,边长与顶点夹角等),当然还包括处理这些属性的方法。该系统中的一切都是发生在名为 VoroDelt 的类中,“类”这个字很好的表达了面向对象的办法,C++ 编程语言中“类”的概念贯穿系统的始终。整个模型都被封装在 VoroDelt 内,它所有的外观显示都是通过一定条件下读取类中数据成员的值实现的,对模型的外部观察决不会影响到类中主要函数的工作。

在确定形成 Delauney 三角形的三个顶点时,程序中实际上是计算某一参考点与其它点的距离,找出离此参考点最近的各点,各自以三点作出三角形,这就保证了在 Delauney 三角形外接圆中不包含任何原始点,也不会形成太大的三角形(大三角形内可能包含其它的原始点)。

用计算机程序做出的二维 Voronoi 模型中的多边形数目是很少的或有限的,这与真实的‘细胞’系统差距甚远,按此初始模型所进行的模拟得出的结果误差可能较大,其结果的可信度会大打折扣,故实际的初始模型均采用周期性结构,以期得到更多数目的多边形‘细胞’。按此方法可在原始网络的周围得到八个同样的复制品结构,使原来初始网络中的多边形数目增加了 8 倍,较好地克服了由于有限多边形‘细胞’数目而引起的仿真结果的误差。形成周期性初始结构的方法描述如下:设模型网络的长为 xlen,宽为 ylen,其面积为 xlen * ylen。在此矩形平面上每布下 1 个点(x_i, y_i)后,再将这个点放到矩形平面以外(即上下左右等位置)。例如($x_i + xlen, y_i$), ($x_i - xlen, y_i$), ($x_i, y_i + ylen$), ($x_i, y_i - ylen$), ($x_i + xlen, y_i + ylen$)……这样相当于有 $n * 9$ 个多边形‘细胞’的‘大’模型在参与该系统仿真,较好的解决了模型较小的问题,但是屏幕上的显示和打印的内容仍然是一个面积为 xlen * ylen 的区域。

了解本程序的创作过程对理解整个模型的结构是有益的。最初的文件有两个,它们是 Core.h 和 Core.cpp,这两个文件是最核心的,里面包含了模型的全部数据结构,自然也包含了类 VoroDelt 的声明与实现。这两个文件是在 Vc++ 6.0 上编译通过的。随后整体模型交互环境的设计在 C++ Builder4.0 上完成,这里包括了数据的输出,图表的显示,流程的控制等工作。

在 C++ Builder4.0 中创建的主要源文件如图 3 所示。

这里有两点需要说明:一是关于 Core.h 与 Core.cpp 的通用性。Core.h 与 Core.cpp 中大部分是标准的 C++ 语言,没有

创建的头文件 .h	创建的源文件 .cpp
About.h	About.cpp
Delauney.h	Delauney.cpp
Chart.h	Chart.cpp
Display.h	Display.cpp
Init.h	Init.cpp
VoronoiForm.h	VoronoiForm. Cpp

图 3 C++ Builder 中创建的主要源文件

使用与特定操作系统相关的特性,唯一使用的两个头文件是 <stdlib.h> 和 <time.h>,主要是在随机函数的调用和随机函数的初始化时需要用到它们。而这两个头文件及文件中定义的大部分常量与函数目前在绝大多数操作系统都是支持的。Core.h 和 Core.cpp 本身已经可以单独调试模型的运作情况了,只需要加上一些打印输出结果的函数即可。并且已经调试成功,Core.h 和 Core.cpp 在 Windows 和 Linux 下编译都可以通过。

在 Windows 下 VC++ 和 Borland C++ 对语句块的定义略有不同,上述 Core.h 与 Core.cpp 最终是在 C++ Builder4.0 下编译通过的,所以这两个文件在 VC 下编译时,编译器会报告变量重复定义的错误。只要按照特定编译器的要求改过就可以了,这不是仿真程序本身的问题。

Borland C++ 与 Linux 下 cc 编译的行为还是比较一致的,Core.h 和 Core.cpp 可以不加任何修改的通过这两种编译器的编译。

在介绍程序的基本框架时,我们觉得采用 Turbo C 设计的系统更加容易理解,下面给出一些我们用到的函数:

```
void dltri1(float i,float il,float j,float jl,float k,float k1) // 由给定的(i,il) 参考点找出形成 Delauney 三角形的另外两个顶点(j,jl) 和(k,k1) //
```

```
void inbox(float i,float il) // 测试(i,il) 给定的点是否在平面上的周期边界框内部 //
```

```
void ptbox(float xc,float yc,float i2,float j2,float k2) // 测试三角形中心点(xc,yc) 是否在平面上的周期边界框内部 //
```

```
void tricen(float x1,float y1,float x2,float y2,float x3, float y3, float xc,float yc) // 计算由点(x1,y1), (x2,y2), (x3,y3) 形成的 Delauney 三角形的中心,中心点坐标是(xc, yc) //
```

```
void vform(float x[nt2],float y[nt2],int nt2, float xlen, float ylen,float perc) // 在网络区域内撒下 nt2 个随机点,点的坐标保存在 x 和 y 两个一维数组中,perc 是一个小数,它限定了点与点之间的最小距离 //
```

```
void vors2() // 通过以上数据形成 Voronoi 网络 //
void vorder(int i, int j - int t) // 由于采用周期边界:一共有九个相同的网络;i 是网络 1 的索引,j 是网络 2 的索引,t = 9,表示有九个相同的网络 //
```

以上简要地描述了几何法产生

(下转第 128 页)

5 总结

本文介绍了目前比较著名的七个网格模拟器 Bricks, ChicSim, EDGSim, GridSim, MicroGrid, OptorSim 和 SimGrid。网格及相关领域的研究者可以根据自己的需要或兴趣选取某些模拟器来参考、研究或使用,甚至还可以参与这些模拟器的开发完善工作。

目前,网格模拟软件还很不成熟。首先,这些模拟器的功能主要集中在网格作业调度算法和数据访问优化算法的评估这两个方面,而资源故障、作业迁移、加密、认证、计费等方面还很少涉及。其次,所模拟的资源仅限于处理器、存储器、内存、网络等,而数据库、科学仪器等资源还未考虑。再次,现有网格模拟器的功能比较简单,对异构资源和跨管理域的模拟能力有限,还无法完成真实复杂的网格模拟。

网格的研究和开发所涵盖的领域非常广。要想在一个网格模拟软件中实现网格所有研究领域的模拟功能非常困难。网格模拟软件的发展刚从简单的单一功能开始。也许在各个网格研究领域的模拟软件逐渐成熟和完善后,可以考虑将它们集成起来,形成一个功能完备的通用的网格模拟系统。

参考文献:

- [1] A Takefusa, S Matsuoka, H Nakada, K Aida and U Nagashima. Overview of a Performance Evaluation System for Global Computing Scheduling Algorithms[C]. Proceedings of 8th IEEE International Symposium on High Performance Distributed Computing (HPDC-8). 1999. 97-104.
- [2] R Buyya and M Murshed. GridSim: a Toolkit for the Modeling and Simulation of Distributed Resource Management and

Scheduling for Grid Computing [J]. The Journal of Concurrency and Computation: Practice and Experience (CCPE), 2002, 14(13-15):1175-1220.

- [3] H Xia, H Dail, H Casanova and A Chien. The MicroGrid: Using Emulation to Predict Application Performance in Diverse Grid Network Environments[C]. Proceedings of the Workshop on Challenges of Large Applications in Distributed Environments (CLADE'04). Honolulu, Hawaii: IEEE Computer Society Press, 2004. 52-63.
- [4] W H Bell, D G Cameron, L Capozza, A P Millar, K Stockinger and F Zini. OptorSim - A Grid Simulator for Studying Dynamic Data Replication Strategies [J]. International Journal of High Performance Computing Applications, 2003, 17(4).
- [5] H Casanova. Simgrid: a Toolkit for the Simulation of Application Scheduling [C]. Proceedings of the First IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2001). Brisbane, Australia: IEEE Computer Society Press, 2001. 430-437.
- [6] H J Song, X Liu, D Jakobsen, R. Bhagwan, X Zhang, K Taura and A Chien. The MicroGrid: a Scientific Tool for Modeling Computational Grids[J]. Scientific Programming, 2000, 8(3): 127-141.

作者简介



李荣胜(1975-),男(壮族),广西大化县人,北京邮电大学电信工程学院博士生,研究方向为网格计算。

徐惠民(1941.5-),男(汉族),上海人,北京邮电大学电信工程学院计算中心主任、教授、博士生导师,研究方向为网格计算。

(上接第91页)

二维 Voronoi 多边形网络的框架,算法是比较容易的,但是该模型的全部数据都是具体仿真实例需要使用的,所以在编程处理 Voronoi 多边形顶点坐标、各边长、各‘细胞’面积、Delauney 三角形顶点坐标、各 Voronoi 多边形的面积、顶点坐标及顶点夹角的计算和保存、以及周期网络的处理方面需要多加小心,以免出错。

4 结束语

以上我们给出了二维 Voronoi 多边形网络的精确模型的实现方法,该模型已被用作为二维肥皂泡的计算机仿真模型,仿真了二维肥皂泡的渐近行为(包括各肥皂泡细胞的面积变化、边长和顶点夹角的变化等)和力学特性,通过计算,我们还可以获得更多的时间演变信息。该仿真模型还可用于其他科研领域,只要在程序中引入反映被仿真系统特性的数学模型,就可能会得到所需要的结果。

参考文献:

— 128 —

- [1] 傅廷亮编著. 计算机模拟技术[M]. 中国科学技术大学出版社,2001.
- [2] D Weaire and J P Kermode. On the distribution of cell areas in a Voronoi network[J]. Philosophical Magazine B, 1986, 53: 101-105.
- [3] 傅廷亮, 傅南枝. 关于随机数及其在细胞结构建模中的应用[J]. 计算机仿真, 2003, 20(9):128-129.
- [4] D Weaire and T L Fu. The Mechanical Behavior of Foams and Emulsions[J]. Journal of Rheology, 1988, 32(3):271-283.
- [5] 傅廷亮, 等. 微电子学与计算机[J]. 1991,8(7)(总第133期):5-8.

作者简介



傅廷亮(1950-),男(汉族),安徽桐城人,副教授,主要研究方向是计算机仿真,计算机图形学和数据库。

尹雪涛(1983.1-),男(汉族),山西太原人,中国科大计算机系学生,主要研究方向是计算机网络安全,数据库和计算机图形学。

张 扬(1976-),男(汉族),山西昔阳人,硕士,研究方向是计算机网络和 VRML 仿真。